An Open Source Toolkit for Word-level Confidence Estimation in Machine Translation

Christophe Servan[†], *Ngoc Tien Le*[†], *Ngoc Quang Luong*[‡], *Benjamin Lecouteux*[†] *and Laurent Besacier*[†]

[†]GETALP – LIG, University of Grenoble Alpes, France
firstname.lastname@imag.fr
[‡]Idiap Research Institute, 1920 Martigny, Switzerland
nluong@idiap.ch

Abstract

Recently, a growing need of Confidence Estimation (CE) for Statistical Machine Translation (SMT) systems in Computer Aided Translation (CAT), was observed. However, most of the CE toolkits are optimized for a single target language (mainly English) and, as far as we know, none of them are dedicated to this specific task and freely available.

This paper presents an open-source toolkit for predicting the quality of words of a SMT output, whose novel contributions are (i) support for various target languages, (ii) handle a number of features of different types (system-based, lexical, syntactic and semantic). In addition, the toolkit also integrates a wide variety of Natural Language Processing or Machine Learning tools to pre-process data, extract features and estimate confidence at word-level. Features for Word-level Confidence Estimation (WCE) can be easily added / removed using a configuration file.

We validate the toolkit by experimenting in the WCE evaluation framework of WMT shared task with two language pairs: French-English and English-Spanish. The toolkit is made available to the research community with ready-made scripts to launch full experiments on these language pairs, while achieving state-of-the-art and reproducible performances.

1. Introduction

Statistical Machine Translation (SMT) has proven its efficiency during the last decade. For Computer Aided Translation (CAT) of documents, the following process is now broadly used: the SMT system produces raw translations then trained professional translators post-edit (correct) translation errors (PE). We believe that this SMT+PE pipeline can be improved using automatic confidence estimation (CE) where the system gives some clues about the quality of the SMT output. For instance, post-editors require to have information about the possible quality of the translation (Should they just post-edit the translation or rewrite the whole output? What are the main words/phrases they need to focus on?).

Building a method that could point out both correct and incorrect parts in SMT output is a key component to solve the above problems. When we limit the concept "parts" to "words", the automatic confidence estimation process is called Word-level Confidence Estimation (WCE).

Past years have seen the emergence of shared tasks to estimate the translation quality (like WMT CE shared task[1]). In 2015, the organizers called for methods to predict the quality of SMT output at run-time on three levels: sentence-level (Task 1), word-level (Task 2) and (new) document-level (Task 3). This paper more precisely deals with the second task (WCE) but we believe it might be of interest to researchers who work in quality assessment for SMT.

Contributions Our experience in participating in *task 2* (WCE) leads us to the following observation: while feature processing is very important to achieve good performance, it requires to call a set of heterogeneous Natural Language Processing tools (for lexical, syntactic, semantic analyses). Thus, we propose to unify the feature processing, together with the call of machine learning algorithms, to facilitate the design of confidence estimation systems. The open-source toolkit proposed (written in *Python* and made available on *GitHub*) integrates some standard as well as in-house features that have proven useful for WCE (based on our experience in WMT 2013 and 2014).

Outline The paper is organized as follow: Section 2 presents WCE task and related works on this topic. Section 3 is an overview of the features we extract while Section 4 describes the toolkit itself. Performances obtained using our WCE toolkit are given in Section 5 while Section 6 illustrates how one can easily apply feature selection for WCE using the provided code. Finally, Section 7 concludes this work and gives some perspectives.

2. WCE formalisation and related work

2.1. WCE formalisation

Machine translation (MT) consists in finding the most probable target language sequence $\hat{e} = (e_1, e_2, ..., e_N)$ given a source language sentence $f = (f_1, f_2, ..., f_M)$. We can represent Word-level Confidence Estimation (WCE) information as a sequence q (same length N of $\hat{e}$) where $q =$

[1] Since 2012 (http://www.statmt.org/wmt12/quality-estimation-task.html)

$(q_1, q_2, ..., q_N)$ and $q_i \in \{good, bad\}$[2]. Basically, the WCE component solves the equation[3]:

$$\hat{q} = \underset{q}{\text{argmax}}\{p(q|f, e)\} \qquad (1)$$

This is a sequence labelling task that can be solved with several Machine Learning techniques such as Conditional Random Fields (CRF) [1]. However, to train sequence labelling models, we need a large amount of training data for which a triplet (f, e, q) is available. In our case, we use binary labels associated to each word: *Good* or *Bad* to indicate whether a word is "correct" or "incorrect", respectively.

2.2. Related work

According to [2], features for Word-level Confidence Estimation (WCE) can be classified in two types regarding their origin: the "external features" and the "internal features". On the one hand, internal features are extracted from the SMT system itself like alignment table, N-best list, word graph, *etc.* On the other hand, external features mainly come from linguistic knowledge sources like syntactic parser, WordNet or BabelNet API, *etc.* In our approach, we use both types of features. They are mostly detailed in Section 3.

The first works about confidence estimation [3, 4], focused at the word level, was inspired by work done in automatic speech recognition [5]. The combination of a large amount of features, through a Naive Bayes model and a Neural Network, showed that Word Posterior Probability (WPP) was the most relevant internal feature. Later on, [6] integrated POS tagging and other external features. In the same way, [7] proposed 70 linguistic features for quality estimation at sentence level. Some of these features can be applied at word level. Their work also revealed the need of efficient machine learning algorithms to integrate multiple features and achieve better performance.

Recent workshops proposed some shared evaluation tasks of WCE systems, in which several attempts of participants to mix internal and external features were successful. The estimation of the confidence score uses mainly classifiers like Conditional Random Fields [8, 9], Support Vector Machines [10] or Perceptron [11].

Further, some investigations were conducted to determine which feature seems to be the most relevant. [10] proposed to filter features using a forward-backward algorithm to discard linearly correlated features. Using Boosting as learning algorithm, [2] was able to take advantage of the most significant features.

Our work, inspired by all those previous papers, proposes to mix internal and external features and uses CRF as decision algorithm to estimate a WCE score. The technical novelty is their integration in a single toolkit, with ready-made scripts, to quickly run reproducible experiments on different language pairs. It also provides a built-in feature selection approach. Contrarily to the toolkit proposed in [12], our framework allows a quick and easy reproduction of the results presented in this paper and addition of new features is straightforward.

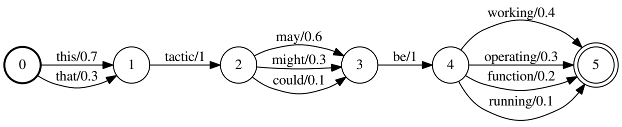

Figure 1: Example of Confusion Network

3. Available Features

Our toolkit extracts several internal and external features to train a classifier, as indicated in Table 1. These features were chosen because of their relevance in previous Word-level Confidence Estimation tasks [13, 14, 15]. Some of them are already described in detail in some previous papers [5, 3, 4, 6, 10, 2, 16]. Consequently, the novel features, which we added into our current toolkit, are in **"bold"** in Table 1. Also, the features in *"italic"* are conventional features but extracted using a new approach.

The feature list could be extended (by us or by other contributors) in the future, since the toolkit is made available to the research community. For instance, we plan to integrate the use of monolingual or bilingual word embeddings following the works of [17].

It is important to note that our toolkit extracts the features regarding *tokens* in the machine translation (MT) hypothesis sentence. In other words, one feature is extracted for each token in the MT output. So, in the Table 1, *target* refers to the feature coming from the MT hypothesis and *source* refers to a feature extracted from the source word aligned to the considered target word. More details on some of these features are given in the next subsections.

3.1. Internal Features

These features are given by the Machine Translation system, which outputs additional data like N-best list.

In addition to features corresponding to source / target words or POS (feat. 5 to 10), **Word Posterior Probability** (WPP), **WPP Max**, **WPP Min** and **Nodes** features are extracted from a confusion network, which comes from the output of the machine translation N-best list. **WPP Exact** is the WPP value for each word concerned at the exact same position in the graph. **WPP Any** extract the same information at any position in the graph. **WPP Min** gives the smallest WPP value concerned by the transition and **WPP Max** its maximum.

In the example shown in Figure 1, the target word *"function"* gets a **WPP Exact** at *0.2*, **WPP Min** at *0.1* and **WPP max** at *0.4*.

[2] q_i could be also more than 2 labels, or even scores but this paper only deals with error detection (binary set of labels).

[3] In the equation, *p* is a probability but it could be any scoring function.

1 *Proper Name*
2 **Unknown Stem**
3 # of Word Occurrences
4 **# of Stem Occurrences**
5 Source POS
6 *Source Word*
7 **Source Stem**
8 Target POS

9 Target Word
10 **Target Stem**
11 *Word context Alignements*
12 *POS context Alignements*
13 **Stem context Alignements**
14 Longest Target N-gram Length
15 Longest Source N-gram Length
16 WPP Exact*

17 WPP Any*
18 WPP Min*
19 WPP Max*
20 Nodes
21 Numerical
22 Punctuation
23 Stop Word
24 Target Backoff Behaviour

25 *Constituent Label*
26 *Distance To Root*
27 *Polysemy Count – Target*
28 **Occur in Bing Translator**
29 Occur in Google Translate

Table 1: Features extracted by the toolkit: highlights in **"bold"** are the new features we propose, the other features are those classically extracted ; we put in *"italic"* those for which we propose a new extraction method compared to previous work (see Section 4.2.3). Features indicated with " * " are internal ones.

3.2. External Features

Below is the list of the external features we use in our toolkit:

- **Proper Name**: indicates if a word is a proper name (same binary features are extracted to know if a token is **Numerical**, **Punctuation** or **Stop Word**).
- **Unknown Stem**: informs whether the stem of the considered word is known or not.
- **Number of Word/Stem Occurrences**: count the occurrences of a word/stem in the sentence.
- **Alignment context features**: these features (#11-13 in Table 1) are based on collocations and proposed by [18]. Collocations could be an indicator for judging if a target word is generated by a particular source word. We also apply the reverse, the collocations regarding the source side:
  - *Source alignment context features*: the combinations of the target word, the source word (with which it is aligned), and one source word before and one source word after (left and right contexts respectively).
  - *Target alignment context features*: the combinations of the source word, the target word (with which it is aligned), and one target word before and one target word after.

  With the example presented in Table 2, the target word "of" is aligned with "de". The source context extracted corresponds to the two words around "de", which are "nature" and "l' ". The *source alignment context features* are "of/nature", "of/de" and "of/l' " In the same way, he *target alignment context features* of "de" are: "de/nature", "de/of" and "de/the".

  We applied the same context extraction for Part-of-Speech and Stems.

Target	the	nature	of	the	independence	granted	...
Source	la	nature	de	l'	indépendance	octroyée	...

Table 2: Example of parallel sentence where words are aligned one-to-one.

- **Longest Target (or Source) N-gram Length**: we seek to get the length $(n+1)$ of the longest left sequence (w_{i-n}) concerned by the current word (w_i) and known by the language model (LM) concerned (source and target sides). For example, if the longest left sequence w_{i-2}, w_{i-1}, w_i appears in the target LM, the longest target n-gram value for w_i will be 3. This value ranges from 0 to the max order of the LM concerned.
- The word's constituent label (**Constituent Label**) and its depth in the constituent tree (**Distance to Root**) are extracted using a syntactic parser, the Figure 2 illustrates the distance between a word and its root in the tree. In the case of *"working"*, the **Constituent Label** is *VBG* and the **Distance to Root** value is *6*.

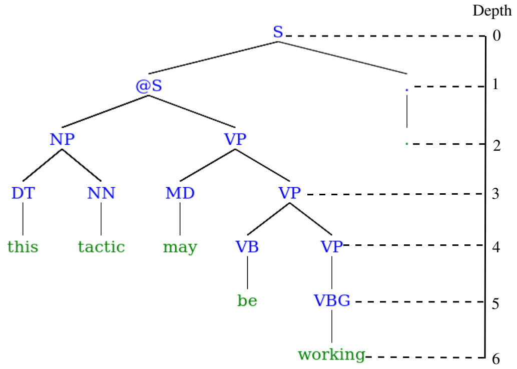

Figure 2: Example of constituent tree.

- **Target Polysemy Count**: we extract the polysemy count, which is the number of meanings of a word in a given language.
- **Occurences in Google Translate** and **Occurences in Bing Translator**: in the translation hypothesis, we (optionnally) test the presence of the target word in online translations given respectively by *Google Translate* and *Bing Translator*.

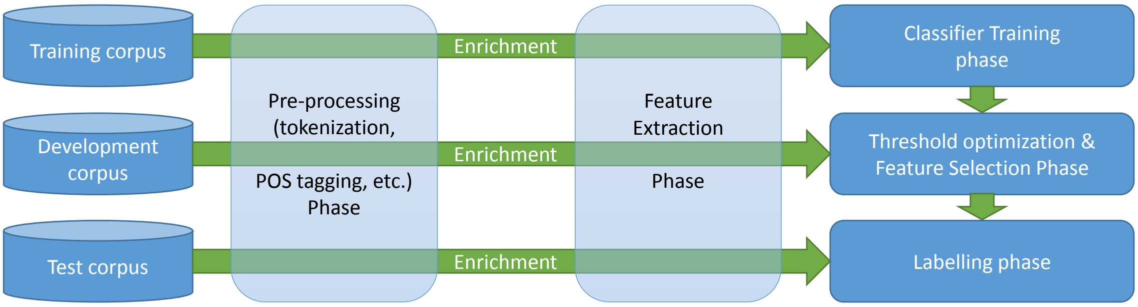
Training corpus
Development corpus
Test corpus
Enrichment
Enrichment
Enrichment
Pre-processing (tokenization, POS tagging, etc.) Phase
Feature Extraction Phase
Classifier Training phase
Threshold optimization & Feature Selection Phase
Labelling phase

Figure 3: Pipeline of our Word-level Confidence Estimation tool

4. Toolkit

In this section, we detail our toolkit, which is a complete out-of-the-box Word-level Confidence Estimation (WCE) system. It is a customizable, flexible, and portable platform.

4.1. Pipeline Overview

Our toolkit is described in Figure 3. It contains three essential components: *preprocessing*, *feature extraction* and *training / labeling*. It integrates several existing Natural Language Processing (NLP) tools and API. It is developed in *Python 3* to use efficiently existing libraries/toolkits as well as being object-oriented designed.

The source code is available on a *GitHub* repository[4] and provided with ready-made scripts to run reproducible experiments on a French–English WCE task (for which the data is also made available).

4.2. System Design

The first steps are the preprocessing and the feature extraction during which the toolkit processes and adds information to the initial corpora available. Then, the most important step consists of training a classifier using the features extracted (training phase) or in the labelling of the test corpus (decoding phase).

We also added a threshold optimization and a feature selection phase which are later described (see Sections 5.5 and 6 respectively for threshold optimization and feature selection).

All these phases can be parameterized using a single configuration file.

4.2.1. *Configuration file*

A configuration file gathers the main WCE parameters. It is stored in YAML[5] format. The main configuration parameters concern the source and target languages involved and the path to the input corpus and its translation.

4.2.2. *Preprocessing Phase*

Preprocessing consists of obtaining POS tags, word alignments and all needed analyses from the available parallel corpus (the target being a MT output made up of raw text – 1-best and N-best of MT). First, input data is lowercased and/or tokenized if necessary. Then, TreeTagger toolkit [19] is applied to get the Part-Of-Speech (POS) tags and stem of each word in both source and target languages. The different POS extracted are normalized. Finally, word alignments are obtained using GIZA++ [20].

4.2.3. *Features Extraction*

As said before, the internal features come from the output of the Statistical Machine Translation (SMT) system. In this part we mainly focus on the extraction of the external features, given by toolkits which are not part of the SMT system.

The TreeTagger toolkit [19] is involved in the extraction of the following features: “Proper Names”, “Unknown Stems” and “Source/Target Stem”. GIZA++ [20] helps us to extract the context alignment features for POS, Word and Stems. To compute the features “Longest Target N-gram Length” and “Longest Source N-gram Length” we use the SRILM toolkit [21]. The word’s constituent label (“Constituent Label”) and its depth in the constituent tree (“Distance to Root”) are also extracted using Bonsai (for French) [22, 23] or Berkeley parser (for other languages) [24]. To represent hierarchical structures and extract the two features, the Natural Language ToolKit (NLTK) [25] in Python is used. The BabelNet [26] API is used to extract the feature “Target polysemy count”.

Finally, the features “Occurences in Google Translate” and “Occurences in Bing Translator” are extracted by using the *Google Translate* and *Bing Translator* API, respectively.

4.2.4. *Training / Decoding Phase*

Once the final feature extraction stage has been completed, we use Conditionnal Random Fields (CRF) as machine learning technique through the Wapiti toolkit [27].

The classifier uses all the chosen features and it is trained on a preliminary labelled corpus (see next section for example of corpora directly usable with our toolkit). During decoding phase, the classifier determines, from a test corpus, whether a word should be labelled as “correct” or “incorrect” (respectively *Good* or *Bad*).

[4] https://github.com/besacier/WCE-LIG
[5] http://www.yaml.org/

5. WCE Experiments

This section presents the experiments done for 2 different language pairs: French–English (*fr–en*) with the corpus provided by [28] and English–Spanish (*en–sp*) corresponding to the WMT shared task on word confidence estimation (2014 edition[6]).

5.1. The French–English post-edited corpus

The *fr–en* corpus contains 10881 translations. It was taken from several French–English news corpora from former WMT evaluation campaigns (from 2006 to 2010) [28].

To obtain the translations, [28] used a French–English phrase-based translation system based on the *Moses* toolkit [29]. This medium-sized system was trained on Europarl and News parallel corpora for a former WMT evaluation shared-task (system more precisely described in [30] - 1.6M parallel sentences and 48M monolingual sentences in target language).

The hypotheses translated were post-edited according to the methodology described in [28]. 10000 random sentences were extracted to create the training data and the remaining sentences were used for the evaluation corpus.

In order to evaluate our Word-level Confidence Estimation (WCE) system, we obtained a sequence q of quality labels (recall that $q = (q_1, q_2, ..., q_N)$ and $q_i \in \{good, bad\}$) using TER-Plus toolkit [31]. Each word or phrase in the hypothesis e_{hyp} is aligned to a word or phrase in the reference (e_{ref}) with different types of edit: "I" (insertions), "S" (substitutions), "T" (stem matches), "Y" (synonym matches), "P" (phrasal substitutions) and "E" (exact match). Then, we recategorize the obtained 6-label set into binary set: the "E", "T" and "Y" belong to the *good* ("G"), whereas the "S", "P" and "I" belong to the *bad* ("B") category.

An example of output of TER-Plus evaluation tool is shown in Table 3.

Original Ref.:	this	is	enough	to	shake	asset	prices	
Original Hyp.:	what	is	enough	to	cower	prices	of	assets
Ref.:	this	is	enough	to	*****	shake	asset	prices
Hyp.:	what	is	enough	to	cower	prices	of	assets
Hyp. After Shift:	what	is	enough	to	cower	of	assets	prices
Alignment:	S	E	E	E	I	S	T	E
Labels:	B	G	G	G	B	B	G	G

Table 3: Example of the TER-Plus toolkit's output processed

5.2. Adaptation to a new language pair

To evaluate our toolkit on another language pair (English–Spanish), we used the official data from WMT 2014 shared task on WCE.

One of the strength of our toolkit is the easiness to adapt it to another language pair within the (so-far) supported languages which are French, English, and Spanish. Thus, a few configuration parameters were changed to move from the French–English (*fr–en*) to English–Spanish (*en–es*), which are mainly the source language, the target language, and paths associated to input files.

Consequently, our WCE toolkit process *en–es* task in the same way as for *fr–en* task, but some features may not be extracted due to language-pair specificities: unavailable tools, no N-best, *etc.* For instance, for the *en–es* task, since the N-best list is not available, we cannot extract the five following internal features: "WPP Exact", "WPP Any", "Nodes", "WPP Min" and "WPP Max".

5.3. Results

The WCE evaluation measures are the Precision (P), the Recall (R) and the F-Measure (F) of each label (as reminder, the decision label can be either *good* or *bad*). We use *wapiti* [27] to train the CRF model and label the words.

5.4. Comparison with the State-of-the-Art

	Systems	*M-F*	F(*bad*)
	FBK-UPV-UEDIN-1 [32]	62.00	48.73
	LIMSI [33]	60.55	47.32
→	*Our toolkit*	*60.76*	*47.17*
	LIG-1 [9]	63.55	44.47
	LIG-2 [9]	63.77	44.11
	FBK-UPV-UEDIN-2 [32]	62.17	42.63

Table 4: Results of the best systems at the Word-level Quality Estimation task (*en–es*) at WMT14 [15], only the Mean F-Measure (*M-F*) and the F-Measure (F) on the bad labels are available to compare the performances of our toolkit.

Using the default decision threshold of our classifier, the Table 4 presents the results obtained in the WMT14 Quality Estimation shared task with the language pair English–Spanish (*en–es*).

The results show that our toolkit obtained similar performances compared to the State-of-the-Art. We could not compare with the CE toolkit mentionned in [12] since they did not provided full results within the framework of the WMT14 evaluation. Future work could involve a comparison between our toolkit and the toolkit presented in [12].

5.5. Decision threshold optimization

Table 5 shows the classification performances of our toolkit for the two different language pairs: the French–English (*fr–en*) and the English–Spanish (*en–es*). The latter corresponds to the Quality Estimation shared task of WMT14 [15].

Our toolkit proposes to optimize the decision threshold but, in this context, what we report can be only considered as an oracle threshold setting since no real development corpus was available for both language pairs. These results are only reported to demonstrate the ability of the toolkit to tune the decision threshold. With this optimization, the scores are improved for the *bad* label (+2.89 points) regarding the results obtained with the default threshold in the *fr–en* task. In the *en–es* task, the oracle threshold sightly improves the results, according to the Mean F-Measure (+0.11 points).

[6] `http://www.statmt.org/wmt14/quality-estimation-task.html`

Task	***Threshold***	***Label***	***P***	***R***	***F***	***M-F***
fr–en	*Default*	*Good*	84.45	90.22	**87.24**	64.96
		Bad	**50.10**	37.16	42.67	
	Optimized	*Good*	**85.60**	85.65	85.62	**65.59**
		Bad	45.61	**45.50**	**45.56**	
en–es (WMT14)	*Default*	*Good*	71.24	**77.73**	**74.35**	60.76
		Bad	**51.82**	43.28	47.17	
	Optimized	*Good*	**71.42**	76.82	74.03	**60.87**
		Bad	51.49	**44.45**	**47.71**	

Table 5: The toolkit's WCE performances with *fr–en* and *en–es* (WMT14) tasks. Note that for each language pair, the first block of results corresponds to the performance obtained with default decision threshold and the second block corresponds to the performance with an oracle threshold (to optimize Mean F-measure of *Good* and *Bad* labels).

6. Features selection

This section illustrates how the toolkit can be used for feature selection and analysis of performance with different feature sets. The next experiments reported were done for the *fr–en* task with the default decision threshold.

6.1. Experimenting with different feature sets

The following feature sets were evaluated in this section:

- the baseline features (*Base.*) given in Table 1 (not **"bold"**, not *"italic"*, no feat. 28-29),
- same as above + modified features estimated with a new method (in *"italic"* in Table 1) are added (*mod.*) ;
- same as above + the new features (*new*) mentionned in Table 1 (the ones in **"bold"**) ;
- same as above + features 28-29 of Table 1 involving online MT systems (*MT*).

Features	***Labels***	***P***	***R***	***F***	***M-F***
Base.	*Good*	81.97	**92.22**	86.80	58.64
	Bad	44.17	23.28	30.48	
+ *mod.*	*Good*	83.21	90.99	86.92	62.00
	Bad	47.24	30.53	37.09	
+ *new*	*Good*	83.55	90.11	86.70	62.65
	Bad	46.75	32.86	38.60	
+ *MT*	*Good*	**84.45**	90.22	**87.24**	**64.96**
	Bad	**50.10**	**37.16**	**42.67**	

Table 6: Improvements obtained regarding the features added. For both labels (Good and Bad) we use the Precision (P), Recall (R) and F-Measure (F). The Mean F-Measure of *Good* and *Bad* labels is presented in the last column.

We can observe for all the steps a general improvement of the Mean F-Measure in Table 6. The baseline is 58.64, while the use of modified features enables to get over 62. The new features show their usefulness with a Mean F-Measure score at 62.65 points. Finally, adding occurences coming from on-line Machine Translation systems enables us to get 64.96 points. Even if using online MT systems for WCE can appear as controversial, this seems to bring useful information to our classifier.

6.2. Feature selection using Sequential Forward Selection (SFS) algorithm

Going further, we propose to process a finer feature selection using the Sequential Forward Selection (SFS) algorithm for which scripts are made available in our toolkit distribution.

While feature selection can be made through several approaches [34], we chose to use the SFS method. It is a bottom up algorithm which starts from a feature set noted Y_k (which can be empty or not) and selects as first feature (x) the one that maximizes the Mean F-Measure, $MF(Y_k + x)$, from a set of features (J_k). The algorithm below summarizes the whole process:

```
while size of $J_k > 0$ do
    $maxval = 0$
    for $x \in J_k$ do
        if $maxval < MF(Y_k + x)$ then
            $maxval \leftarrow MF(Y_k + x)$
            $bestfeat \leftarrow x$
        end if
    end for
    add $bestfeat$ to $Y_k$
    remove $bestfeat$ from $J_k$
end while
```

In Table 7 we present the result of the SFS algorithm, which ranks our new features starting from an empty feature set. The dash line marks the limit of the best feature set according to the Mean F-Measure (with 65.14 points).

It appears that most of new features we added (in **"bold"**) bring relevant information associated to classical ones (no highlight and in *"italic"*). Only the feature "Target Stem" seems to be irrelevant for the *fr–en* task. One reason for that might be that for the English language, stem and words features may be highly correlated.

Rank	Feature	Rank	Feature
1	**Stem context Alignements**	16	Stop Words
2	WPP Exact	17	Nodes
3	*Word context Alignements*	18	**# of Stem Occurrences**
4	WPP Max	19	Numeric
5	WPP Any	20	**Unknown Stem**
6	WPP Min	21	Target Word
7	*POS context Alignements*	22	Source POS
8	Occur in Google Translate	23	*Polysemy Count – Target*
9	Longest Target N-gram Length	24	*Source Word*
10	**Occur in Bing Translate**	25	*Constituent Label*
11	**Source Stem**	26	Punctuation
12	Target Backoff Behaviour	27	**Target Stem**
13	Longest Source N-gram Length	28	*Proper Name*
14	# of Word Occurrences	29	Target POS
15	*Distance To Root*		

Table 7: Rank of each feature according to the Sequential Forward Selection algorithm within the framework of the *fr–en* task. The Dash line marks the best Mean F-Measure score obtained with 65.14 points.

This feature selection functionnality is provided with the toolkit, which means that whatever set of features the user wants to test, he/she can apply the SFS algorithm very easily.

7. Conclusion and Perspectives

This paper presented our Word Confidence Estimation (WCE) approach made available through an open-source toolkit. It combines some classical features as well as some new in-house features. All these features are passed through a Conditional Random Fields (CRF) classifier to estimate the correctness of a word.

The WCE experiments conducted achieve State-of-the-Art and reproducible performances measured on two different data sets corresponding to two language pairs (French–English and English–Spanish). Thanks to its flexibility, our toolkit is nearly language independent, as long as the user can provide grammars and models for the specified languages.

Our WCE toolkit has been packaged and released for others to be able to reproduce rapidly the experiments reported in this article. This package is made available on a *GitHub* repository[7] under the licence GPL V3.

In addition to this toolkit, comes a special module, which enables feature selection automatically using SFS algorithm (sequential forward selection). A more performant algorithm will be added in the near future like the Sequential Floating Forward Selection algorithm, which has backtracking capabilities.

Further work will focus on (i) adding features (based on word embeddings for instance) and (ii) evaluating the toolkit efficiency in a real Computer Assisted Translation (CAT) framework. We also plan to extend our toolkit to the design of WCE for speech recognition and speech translation tasks.

8. Acknowledgement

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper.

This work was partially founded by the French National Research Agency (ANR) through the KEHATH Project.

9. References

[1] J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," in *Proceedings of 18th International Conference on Machine Learning*. Morgan Kaufmann, San Francisco, Californie, États-Unis d'Amrique, 2001, pp. 282–289.

[2] N.-Q. Luong, L. Besacier, and B. Lecouteux, "Towards Accurate Predictors of Word Quality for Machine Translation: Lessons Learned on French - English and English - Spanish Systems," *Data and Knowledge Engineering*, p. 11, Apr. 2015.

[3] N. Ueffing, K. Macherey, and H. Ney, "Confidence Measures for Statistical Machine Translation," in *Proceedings of the MT Summit IX*, New Orleans, LA, September 2003, pp. 394–401.

[4] J. Blatz, E. Fitzgerald, G. Foster, S. Gandrabur, C. Goutte, A. Kulesza, A. Sanchis, and N. Ueffing, "Confidence estimation for machine translation," in *Proceedings of COLING 2004*, Geneva, April 2004, pp. 315–321.

[5] F. Wessel, R. Schlüter, K. Macherey, and H. Ney, "Confidence Measures for Large Vocabulary Continuous Speech Recognition," *IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING*, 2001.

[6] D. Xiong, M. Zhang, and H. Li, "Error Detection for Statistical Machine Translation Using Linguistic Features," in *Proceedings of the 48th Association for Computational Linguistics*, Uppsala, Sweden, July 2010, pp. 604–611.

[7] M. Felice and L. Specia, "Linguistic Features for Quality Estimation," in *Proceedings of the 7th Workshop on Statistical Machine Translation*, Montreal, Canada, June 7-8 2012, pp. 96–103.

[8] A. L.-F. Han, Y. Lu, D. F. Wong, L. S. Chao, L. He, and J. Xing, "Quality estimation for machine translation using the joint method of evaluation criteria and statistical modeling," in *Proceedings of the Eighth Workshop on Statistical Machine Translation*, Sofia, Bulgaria, Aug. 2013, pp. 365–372.

[9] N.-Q. Luong, L. Besacier, and B. Lecouteux, "LIG System for Word Level QE task at WMT14," in *Proceedings of the Ninth Workshop on Statistical Machine Translation*, Baltimore, Maryland USA, June 2014, pp. 335–341.

[10] D. Langlois, S. Raybaud, and K. Smaïli, "Loria system for the WMT12 quality estimation shared task," in *Proceedings of the Seventh Workshop on Statistical Machine Translation*, Baltimore, Maryland USA, June 2012, pp. 114–119.

[11] E. Bicici, "Referential Translation Machines for Quality Estimation," in *Proceedings of the Eighth Workshop on Statistical Machine Translation*. Sofia, Bulgaria: Association for Computational Linguistics, August 2013, pp. 343–351.

[12] L. Specia, G. Paetzold, and C. Scarton, "Multi-level Translation Quality Prediction with QuEst++," in *53rd Annual Meeting of the Association for Computational Linguistics and Seventh International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing: System Demonstrations*, 2015, pp. 115–120.

[13] C. Callison-Burch, P. Koehn, C. Monz, M. Post, R. Soricut, and L. Specia, "Findings of the 2012 Workshop on Statistical Machine Translation," in *Proceedings of the Seventh Workshop on Statistical Machine*

[7] https://github.com/besacier/WCE-LIG

Translation. Montréal, Canada: Association for Computational Linguistics, June 2012, pp. 10–51.

[14] O. Bojar, C. Buck, C. Callison-Burch, C. Federmann, B. Haddow, P. Koehn, C. Monz, M. Post, R. Soricut, and L. Specia, "Findings of the 2013 Workshop on Statistical Machine Translation," in *Proceedings of the Eighth Workshop on Statistical Machine Translation*. Sofia, Bulgaria: Association for Computational Linguistics, August 2013, pp. 1–44.

[15] O. Bojar, C. Buck, C. Federmann, B. Haddow, P. Koehn, J. Leveling, C. Monz, P. Pecina, M. Post, H. Saint-Amand, R. Soricut, L. Specia, and A. Tamchyna, "Findings of the 2014 Workshop on Statistical Machine Translation," in *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Baltimore, Maryland, USA: Association for Computational Linguistics, June 2014, pp. 12–58.

[16] S. Raybaud, D. Langlois, and K. Smaili, ""This sentence is wrong." Detecting errors in machine-translated sentences," *Machine Translation*, vol. 25, no. 1, pp. p. 1–34, Aug. 2011.

[17] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," in *NIPS*, 2013.

[18] N. Bach, F. Huang, and Y. Al-Onaizan, "Goodness: A Method for Measuring Machine Translation Confidence," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, Portland, Oregon, June 2011, pp. 211–219.

[19] H. Schmid, "Improvements in Part-of-Speech Tagging with an Application to German," in *Proceedings of the ACL SIGDAT-Workshop*, Dublin, Ireland, 1995.

[20] F. J. Och and H. Ney, "A Systematic Comparison Of Various Statistical Alignment Models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[21] A. Stolcke, "SRILM - An Extensible Language Modeling Toolkit," in *Seventh International Conference on Spoken Language Processing*, Denver, USA, 2002, pp. 901–904.

[22] A. Laurent, N. Camelin, and C. Raymond, "Boosting bonsai trees for efficient features combination: application to speaker role identification," in *InterSpeech*, Singapour, September 2014, pp. 76–80.

[23] M. Candito, J. Nivre, P. Denis, and E. H. Anguiano, "Benchmarking of Statistical Dependency Parsers for French," in *Proceedings of COLING'2010*, 2010.

[24] S. Petrov and D. Klein, "Improved Inference for Unlexicalized Parsing," in *HLT-NAACL*, 2007.

[25] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*. O'Reilly Media Inc, 2009.

[26] R. Navigli and S. P. Ponzetto, "BabelNet: The Automatic Construction, Evaluation and Application of a Wide-Coverage Multilingual Semantic Network," *Artificial Intelligence*, vol. 193, pp. 217–250, 2012.

[27] T. Lavergne, O. Cappé, and F. Yvon, "Practical Very Large Scale CRFs," in *Proceedings the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*. Association for Computational Linguistics, July 2010, pp. 504–513.

[28] M. Potet, E. Esperança-Rodier, L. Besacier, and H. Blanchon, "Collection of a Large Database of French-English SMT Output Corrections," in *Proceedings of the eighth international conference on Language Resources and Evaluation (LREC)*, Istanbul, Turkey, May 2012.

[29] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: Open source toolkit for statistical machine translation," in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, Prague, Czech Republic, June 2007, pp. 177–180.

[30] M. Potet, L. Besacier, and H. Blanchon, "The LIG machine translation system for WMT 2010," in *Proceedings of the joint fifth Workshop on Statistical Machine Translation and Metrics MATR (WMT2010)*, A. Workshop, Ed., Uppsala, Sweden, 11-17 July 2010.

[31] M. Snover, N. Madnani, B. Dorr, and R. Schwartz, "TERp system description," in *MetricsMATR workshop at AMTA*, 2008.

[32] J. G. Camargo de Souza, J. González-Rubio, C. Buck, M. Turchi, and M. Negri, "FBK-UPV-UEdin participation in the WMT14 Quality Estimation shared-task," in *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Baltimore, Maryland, USA: Association for Computational Linguistics, June 2014, pp. 322–328.

[33] G. Wisniewski, N. Pécheux, A. Allauzen, and F. Yvon, "LIMSI Submission for WMT'14 QE Task," in *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Baltimore, Maryland, USA: Association for Computational Linguistics, June 2014, pp. 348–354.

[34] M. Kudo and J. Sklansky, "Comparison of algorithms that select features for pattern classifiers," *Pattern recognition*, vol. 33, no. 1, pp. 25–41, 2000.